

REMARKS

Reconsideration and allowance of this subject application are respectfully requested.

Applicant has made a claim for priority under 35 U.S.C. §119 based on five United Kingdom applications listed on the cover sheet filed with this application and in the submission made October 11, 2001 providing certified copies of each one of those foreign applications. Acknowledgement of the claim for foreign priority is requested as well as confirmation of receipt of the certified copies of the priority documents.

The Examiner has acknowledged consideration of the references cited in the Information Disclosure filed on November 27, 2001. Applicant recently filed an Information Disclosure Statement on March 31, 2004. Acknowledgement that those references have been considered and return of an initialed PTO-1449 form are respectfully requested.

Claims 1-4 and 10-16 stand rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent 6,351,844 to Bala. This rejection is respectfully traversed.

To establish that a claim is anticipated, the Examiner must point out where each and every limitation in the claim is found in a single prior art reference. *Scripps Clinic & Research Found. v. Genentec, Inc.*, 927 F.2d 1565 (Fed. Cir. 1991). Every limitation contained in the claims must be present in the reference, and if even one limitation is missing from the reference, then it does not anticipate the claim. *Kloster Speedsteel AB*

v. Crucible, Inc., 793 F.2d 1565 (Fed. Cir. 1986). Bala fails to satisfy this rigorous standard.

Independent claims 1, 15, and 20 are directed to the problem of providing a more efficient way of determining whether a non-native subroutine was called from a native calling program or a non-native calling program. The known method of performing this determination is for the return instruction at the end of each non-native subroutine to test, prior to making a return, whether that return is to a native calling program or a non-native calling program. In contrast, claims 1, 15, and 20 provide specific types of return instruction that effectively code into the non-native subroutine the nature of the return to be made.

As one non-limiting embodiment, the Examiner's attention is directed to the example illustrated in Figure 19. A veneer, non-native subroutine 114 is provided between the native calling program and the non-native subroutine. At the end of the non-native subroutine, a return to non-native instruction is executed which either returns processing to a non-native calling program or to a non-native veneer subroutine, depending from where the subroutine was called. The common non-native subroutine saves memory space and also provides the flexibility to return either to a native instruction or to a non-native instruction without having to support the additional overhead of checking for the nature of the calling program each time a return instruction is encountered.

Although Bala describes a dynamic translator for translating of non-native code into native code for execution by the processor, Bala is directed to a *different problem* of selecting regions of non-native code for storage as native code in a code cache. The selection strategy increases the time spent executing from the code cache. Bala neither discloses nor suggests determining whether a non-native subroutine was called from a native calling program or a non-native calling program. Bala merely discloses how control is passed between an instruction interpreter and a translated code cache depending on a cache hit/miss result.

The Examiner asserts that column 3, lines 5-53 of Bala discloses that the instruction translator is responsive to a return to non-native instruction of the non-native instruction set to return processing to a non-native instruction, as recited in each independent claim. Applicant respectfully disagrees. Here, Bala merely specifies that when a translated code segment (native code segment) corresponding to the non-native code to be executed is found not to be present in the translated code cache, (i.e., if a cache "miss" occurs), then program execution branches to an untranslated instruction address. Bala's interpreter does not respond to an actual "return to non-native instruction" to return processing from non-native instructions to a further non-native instruction. Rather, Bala's interpreter responds to a cache miss to switch from processing native instructions to processing non-native instructions.

The Examiner further asserts that column 13, lines 50-59 discloses that Bala's translator responds to a "return to native" instruction of the non-native instruction set to

return processing to a native instruction. But this portion of column 13 merely discloses that for a non-native instruction corresponding to a particular address, there exists corresponding translated code in the translated code cache so that a cache hit occurs, and control is transferred to the translated code cache. Thus, native instructions corresponding to the particular non-native instruction are executed directly from cache. Control remains with the translated code cache until an untranslated address is encountered, whereupon a cache miss will occur, and control is returned to the interpreter so that the non-native code is dynamically translated into native program code for execution by the processor core. Thus, for a given address, the Bala's interpreter responds to a cache hit to switch control to native instructions resident in the translated code cache and responds to a cache miss to return control from the cache to the interpreter. Bala's interpreter does not respond to dedicated "return to native instruction" to return processing from non-native instructions to a native instruction. Bala's interpreter is responsive to a cache hit to switch from processing non-native instructions to native instructions.

Contrary to the Examiner's assertion, Bala does not disclose an instruction translator responsive to both a specific "return to non-native" instruction and a specific "return to native" instruction of a non-native instruction set. Lacking multiple claim features, the rejection of claims based on Bala should be withdrawn.

Although the Examiner rejects claim 5 under 35 U.S.C. §103 as unpatentable over Bala in view of U.S. Patent 5,875,336 to Dickol et al and claims 6-9 as being

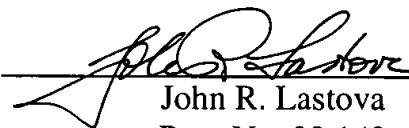
NEVILL
Appl. No. 09/887,561
April 7, 2004

unpatentable over Bala in view of U.S. Patent 6,091,897 to Yates et al., neither of these references overcome the deficiencies of Bala with respect to the independent claims.

This application is in condition for allowance. An early notice to that effect is earnestly notice.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: 
John R. Lastova
Reg. No. 33,149

JRL:at
1100 North Glebe Road, 8th Floor
Arlington, VA 22201-4714
Telephone: (703) 816-4000
Facsimile: (703) 816-4100